© MAY 2026 | IRE Journals | Volume 9 Issue 11 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I11-1717906

IoT-Based Smart Soil Quality Monitoring and Classification for Sustainable Agriculture

RISHAB NS[1], PROF. RAKSHITHA B. S[2]
[1]*Student, Department of Computer Science &IT Jain (Deemed-To-Be-University) Bangalore, India*
[2]*Assistant Professor Department of Computer Science &IT Jain (Deemed-To-Be-University) Bangalore, India*

Abstract- Precision agriculture has become an essential approach for improving crop productivity and sustainable farming practices through intelligent monitoring systems. Real-time monitoring of soil parameters such as moisture, temperature, pH, and nutrient content enables farmers to make informed irrigation and fertilisation decisions. This paper proposes an IoT-based smart soil quality monitoring and classification framework using ESP32 edge devices, soil sensors, and Machine Learning techniques for sustainable agriculture. The system integrates soil moisture and environmental sensors with predictive analytics for real-time soil monitoring. Data preprocessing techniques are applied to remove noisy values and improve prediction accuracy. Random Forest, Decision Tree, and Logistic Regression algorithms are implemented for soil condition classification. The proposed system improves irrigation efficiency, reduces water wastage, and supports sustainable agriculture practices.

Keywords— Precision Agriculture, IoT, ESP32, Soil Moisture Monitoring, Machine Learning, Random Forest, Sustainable Agriculture.

I. INTRODUCTION

1.1 Background of the Study

Agriculture is one of the most important sectors supporting food production and economic development. Traditional farming methods often rely on manual inspection of soil conditions, leading to inefficient irrigation and improper resource utilisation. With advancements in Internet of Things (IoT) technology and Machine Learning, intelligent monitoring systems have become more practical and efficient [1]. Precision agriculture uses sensors, communication systems, and intelligent algorithms to continuously monitor environmental conditions and optimise farming activities. Soil parameters such as moisture, temperature, humidity, pH, and nutrient content directly influence crop growth and productivity. Accurate monitoring of these parameters helps reduce water wastage and improve crop yield [2]. The proposed research develops an IoT-based smart soil quality monitoring and classification system using ESP32, Soil Moisture Sensor, DHT11 sensor, and Machine Learning algorithms for sustainable agriculture applications.

1.2 Problem Statement

Current soil monitoring systems suffer from several limitations that reduce the efficiency of precision agriculture systems. Existing monitoring devices often produce inaccurate readings due to environmental noise, sensor degradation, and improper calibration. Most traditional systems are reactive in nature and provide only current sensor values without intelligent prediction or automated recommendations.

The major problems identified are:

- Sensor degradation in harsh soil conditions.
- Noisy and inaccurate sensor readings.
- Lack of automated decision-making.
- High maintenance requirements.
- Limited integration of Machine Learning techniques.

These issues can lead to improper irrigation decisions, fertiliser wastage, and reduced crop productivity.

1.3 Motivation

The growing demand for sustainable agriculture and efficient resource management has increased the importance of intelligent farming technologies. Water scarcity, climate change, and increasing food demand have created significant challenges for modern agricultural systems. In many regions, inefficient irrigation practices result in excessive water consumption, soil degradation, and reduced crop productivity. Therefore, there is a strong need for low-cost and intelligent monitoring systems capable of improving irrigation management and supporting sustainable farming practices. Recent advancements in Internet of Things (IoT), embedded systems, and Machine Learning have created opportunities for developing smart agricultural monitoring systems capable of real-time data collection and predictive analysis. IoT sensors can continuously monitor soil and environmental conditions, while Machine Learning algorithms can analyse sensor data and generate intelligent predictions regarding soil health and irrigation requirements. Integrating these technologies can help farmers make data-driven decisions instead of relying solely on manual observation and traditional methods. The motivation behind this research is to develop a practical and beginner-friendly smart soil monitoring system capable of continuously analysing soil conditions using real-time sensor data. The proposed framework combines ESP32, soil moisture sensors, DHT11 environmental sensors, and Machine Learning algorithms to create a predictive monitoring solution for sustainable agriculture. The system aims to reduce water wastage, improve irrigation efficiency, enhance crop productivity, and provide real-time soil health recommendations.

1.4 Objectives of the Study

The primary objectives of this research are:

1. To design and implement an IoT-based soil monitoring system using ESP32 and soil sensors.
2. To collect and preprocess soil sensor data using data cleaning techniques.
3. To classify soil conditions using Machine Learning algorithms.
4. To provide real-time recommendations regarding irrigation and soil health.
5. To develop a simple and low-cost prototype suitable for sustainable agriculture.

1.5 Contributions of the Paper

This research contributes to the field of smart agriculture by proposing an IoT-based soil quality monitoring and classification framework that combines real-time sensing, data preprocessing, and Machine Learning techniques for intelligent irrigation management. The proposed system demonstrates how low-cost embedded hardware and predictive analytics can be integrated to improve agricultural efficiency and support sustainable farming practices.

The major contributions of this research are summarised as follows:

1. Development of an IoT-based smart soil monitoring system using ESP32, Soil Moisture Sensor, and DHT11 environmental sensor for real-time data acquisition.
2. Integration of data cleaning and preprocessing techniques to improve the quality and reliability of sensor data before Machine Learning analysis.
3. Implementation of Machine Learning algorithms, including Random Forest, Decision Tree, and Logistic Regression, for soil condition classification.
4. Real-time classification of soil conditions into Dry Soil, Healthy Soil, and Excess Water categories based on live sensor readings.
5. Design of a beginner-friendly and low-cost prototype suitable for practical agricultural applications and academic research purposes.
6. Generation of real-time monitoring outputs and irrigation recommendations using Arduino IDE Serial Monitor.
7. Development of an ML-ready sensor dataset capable of supporting future predictive agriculture research and intelligent irrigation systems.
8. Provision of a scalable framework that can be extended with advanced sensors, cloud computing, AI-based recommendations, and automated irrigation mechanisms in future work.

II. LITERATURE REVIEW

Recent advancements in precision agriculture have significantly improved the development of intelligent soil monitoring systems using Internet of Things (IoT) technologies, embedded systems, and Machine Learning algorithms. Researchers have explored various sensor-based monitoring frameworks using platforms such as Arduino, Raspberry Pi, ESP32, and cloud computing systems for real-time agricultural monitoring and predictive analysis.

1. Shubha B. et al. proposed a Raspberry Pi-based soil monitoring system capable of continuously monitoring environmental conditions and soil

moisture levels with approximately 94% monitoring accuracy [1].
2. Dr. R. Kalpana developed a Machine Learning-based irrigation recommendation framework using the Random Forest algorithm for intelligent water management and fertiliser optimisation [2].
3. M. Rajkumar implemented CNN and MLP models for image-based soil classification and nutrient prediction. The proposed framework improved classification accuracy but required high computational resources [3].
4. S. J. Miller proposed an edge-based agricultural monitoring framework using TensorFlow Lite for low-latency Machine Learning predictions on embedded systems [4].

Recent research has also explored the integration of advanced artificial intelligence techniques such as Agentic AI and Retrieval-Augmented Generation (RAG) systems for automated agricultural recommendations and intelligent remediation planning. These systems aim to provide adaptive decision-making capabilities for precision agriculture environments.

III. PROPOSED METHODOLOGY

3.1 System Architecture

The proposed system is designed as an IoT-based smart soil quality monitoring and classification framework capable of performing real-time environmental sensing, data preprocessing, and Machine Learning-based soil condition prediction. The architecture integrates ESP32, Soil Moisture Sensor, DHT11 environmental sensor, and Machine Learning models to create an intelligent agricultural monitoring system for sustainable farming applications. The overall framework consists of four major layers: the Perception Layer, Communication Layer, Processing Layer, and Application Layer. Each layer performs a specific function within the monitoring and prediction workflow.

Perception Layer

The Perception Layer is responsible for collecting environmental and soil-related data using IoT sensors. The hardware components used in this layer include:

- ESP32 Development Board
- Soil Moisture Sensor
- DHT11 Temperature and Humidity Sensor

The Soil Moisture Sensor continuously measures the water content present in the soil, while the DHT11 sensor measures environmental temperature and humidity conditions. These sensors provide real-time analogue and digital sensor readings to the ESP32 microcontroller.

Communication Layer

The Communication Layer enables data transmission between the ESP32 and the monitoring system. The ESP32 acts as the primary controller and processes incoming sensor values before transmitting the collected data to the connected monitoring interface through USB communication. The system can also be extended for wireless WiFi-based cloud communication in future implementations.

Processing Layer

The Processing Layer performs data preprocessing, data cleaning, and Machine Learning-based analysis. Sensor readings collected from the IoT devices often contain noisy values, duplicate records, or abnormal sensor fluctuations caused by environmental disturbances. Therefore, preprocessing techniques such as duplicate removal, missing value handling, normalisation, and noise filtering are applied before training the Machine Learning models. The processed dataset is then used for classification using Machine Learning algorithms, including Random Forest, Decision Tree, and Logistic Regression. These algorithms analyse the sensor values and classify the soil condition into predefined categories.

Application Layer

The Application Layer displays the final output and irrigation recommendations based on the Machine Learning predictions. The output is displayed in the Arduino IDE Serial Monitor and categorises soil conditions into:

- Dry Soil
- Healthy Soil
- Excess Water

The system also provides real-time recommendations such as “Water Needed” or “Stop Watering” depending on the detected soil condition. The proposed methodology combines IoT sensing, real-time monitoring, data cleaning, and predictive analytics into a unified framework capable of improving irrigation efficiency and supporting sustainable agricultural practices.

3.2 Hardware Components

The proposed smart soil monitoring system uses low-cost IoT hardware components for real-time environmental sensing, data acquisition, and predictive analysis. The selected hardware components are lightweight, energy-efficient, and suitable for beginner-friendly smart agriculture implementations. The ESP32 microcontroller acts as the primary processing unit and communicates with the connected sensors to collect and analyse soil and environmental parameters. The Soil Moisture Sensor is used to measure the amount of water present in the soil by detecting variations in electrical conductivity between the sensor probes. The DHT11 sensor measures environmental temperature and humidity conditions, which are useful for understanding the surrounding agricultural environment. These sensor readings are processed by the ESP32 and transmitted to the monitoring system for Machine Learning-based classification and real-time output generation. The hardware setup was selected to ensure simplicity, affordability, scalability, and compatibility with IoT-based agricultural monitoring systems. The overall implementation supports real-time soil condition classification and can be extended with additional sensors and cloud-based communication modules in future work.

TABLE.1. HARDWARE COMPONENTS USED IN THE PROPOSED SYSTEM

Component	Purpose
ESP32 Development Board	Main controller and data processing unit
Soil Moisture Sensor	Measures soil water content
DHT11 Sensor	Measures temperature and humidity
Jumper Wires	Hardware connections
USB Cable	Power supply and programming
Laptop	Data monitoring and ML implementation

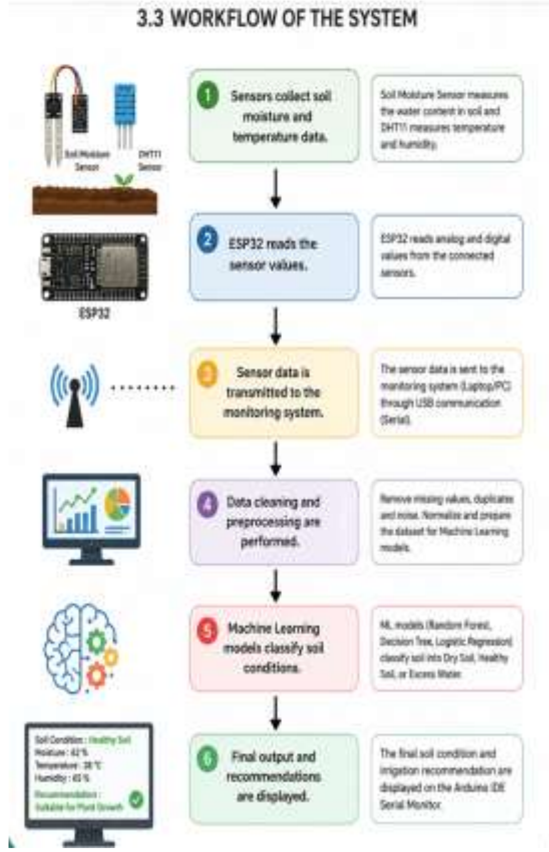

Fig. 1. Workflow of the Proposed Smart Soil Monitoring System

IV. DATASET DESCRIPTION

4.1 Data Source

The dataset used in this research was generated using real-time sensor readings collected from the implemented IoT-based soil monitoring system. The data was obtained using the Soil Moisture Sensor and DHT11 Temperature and Humidity Sensor connected to the ESP32 microcontroller. The sensors continuously monitored environmental and soil conditions under different moisture levels to create a dataset suitable for Machine Learning-based soil classification. The collected dataset consists of soil moisture readings, temperature values, humidity values, and corresponding soil condition labels. The dataset was generated during experimental testing by exposing the sensors to different soil conditions such as dry soil, normal moisture soil, and excess water conditions. These sensor readings were transmitted to the monitoring system through the ESP32 and stored for preprocessing and Machine Learning analysis.

The dataset used in this research consists of:

- Soil moisture readings
- Temperature values
- Humidity values
- Soil condition labels

The collected data was later cleaned, normalised, and prepared for Machine Learning model training and testing. The generated dataset supports real-time classification of soil conditions into Dry Soil, Healthy Soil, and Excess Water categories for intelligent irrigation management and sustainable agriculture applications.

4.2 Dataset Features

The dataset generated for this research contains environmental and soil-related parameters collected from IoT sensors connected to the ESP32 microcontroller. These attributes are used as input features for preprocessing, Machine Learning model training, and soil condition classification. Each attribute represents an important agricultural parameter that influences soil health and irrigation requirements. The Soil Moisture Sensor provides analogue readings that indicate the amount of water present in the soil, while the DHT11 sensor measures environmental temperature and humidity conditions. Based on these sensor readings, the soil condition is classified into predefined categories such as Dry Soil, Healthy Soil, and Excess Water.

TABLE.2 DATASET FEATURES USED FOR MACHINE LEARNING

Attribute Name	Description
Moisture Value	Soil moisture sensor reading
Temperature	Environmental temperature in °C
Humidity	Environmental humidity percentage
Soil Condition	Classified soil category

The moisture value acts as the primary feature for soil condition prediction, while temperature and humidity values help improve the reliability and contextual understanding of the environmental conditions. The Soil Condition attribute represents the target output used for Machine Learning classification. Sample dataset values collected during experimental testing are presented below:

TABLE.3. SAMPLE DATASET VALUES

Moisture Value	Temperature	Humidity	Soil Condition
3500	34°C	42%	Dry Soil
2200	29°C	65%	Healthy Soil
1200	25°C	82%	Excess Water

These dataset attributes are further processed during the data cleaning and preprocessing stage before applying Machine Learning algorithms for predictive soil classification.

V. DATA CLEANING AND PREPROCESSING

5.1 Need for Data Cleaning

Raw sensor data collected from IoT devices often contains errors caused by environmental disturbances, unstable sensor readings, hardware limitations, and communication fluctuations. These issues reduce the accuracy of Machine Learning predictions and negatively affect the reliability of smart agriculture monitoring systems. Since the proposed framework depends on real-time sensor readings obtained from the Soil Moisture Sensor and DHT11 sensor, proper data cleaning and preprocessing are necessary before implementing Machine Learning algorithms. During experimental testing, several inconsistencies were observed in the collected sensor dataset. Moisture values occasionally fluctuated because of unstable probe contact with soil samples. Similarly, temperature and humidity readings sometimes produced abnormal values during rapid environmental changes. Duplicate records were also generated because the ESP32 continuously transmitted sensor values through serial communication. If these noisy and inconsistent readings are directly used for Machine Learning training, the classification accuracy of the proposed system may decrease significantly.

The common issues observed in the collected sensor dataset include:

- Missing values

© MAY 2026 | IRE Journals | Volume 9 Issue 11 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I11-1717906

- Duplicate records
- Noisy sensor readings
- Invalid moisture values
- Abnormal temperature readings
- Unstable humidity fluctuations

For example, moisture readings containing NULL values and unrealistic temperature values below 0°C were removed from the dataset during preprocessing. Duplicate records generated during continuous serial monitoring were also eliminated to avoid biased predictions. Therefore, data cleaning and preprocessing are essential stages in the proposed research framework before training and testing the Machine Learning models. Proper preprocessing improves prediction reliability, enhances model performance, and supports accurate soil condition classification for intelligent irrigation management.

5.2 Data Cleaning Techniques Used

The following preprocessing methods are applied:

Removal of Missing Values

Sensor readings containing missing or NULL values were removed from the dataset because incomplete records can negatively affect Machine Learning prediction accuracy. Missing values were mainly caused by temporary communication interruptions and unstable sensor responses.

Duplicate Record Elimination

During continuous serial communication between the ESP32 and the monitoring system, duplicate sensor entries were occasionally generated. Duplicate records were identified and removed to avoid biased model training and repeated data patterns.

Noise Filtering

Unstable sensor readings and abnormal spikes caused by environmental disturbances were filtered using threshold-based validation methods. Extremely high or unrealistic values were removed from the dataset to improve stability and consistency.

Data Normalization

The collected sensor values were normalised into a consistent range before Machine Learning implementation. Normalisation improves model performance by preventing large numerical differences between sensor features.

Label Encoding

The soil condition categories, such as Dry Soil, Healthy Soil, and Excess Water, were converted into numerical labels for Machine Learning classification.

TABLE.4. PREPROCESSING TECHNIQUES USED

Preprocessing Technique	Purpose
Missing Value Removal	Removes incomplete records
Duplicate Removal	Eliminates repeated entries
Noise Filtering	Removes unstable sensor values
Normalization	Scales sensor data consistently
Label Encoding	Converts labels into numeric form

TABLE.5. DATASET BEFORE AND AFTER PREPROCESSING

Before Preprocessing	After Preprocessing
Moisture = NULL	Removed
Duplicate Record	Removed
Temperature = -4°C	Removed
Moisture = 3500	Retained
Moisture = 2200	Retained

The cleaned dataset generated after preprocessing was then used for training and testing the Random Forest, Decision Tree, and Logistic Regression Machine Learning models implemented in the proposed smart soil monitoring system.

5.3 Example of Data Cleaning

TABLE VI. SAMPLE DATA CLEANING RESULTS

Before Cleaning

Moisture Value	Temperature	Humidity	Output
-50	28°C	61%	Dry Soil
5000	NULL	72%	Healthy

© MAY 2026 | IRE Journals | Volume 9 Issue 11 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I11-1717906

			Soil
2300	29°C	65%	Healthy Soil
1200	-3°C	85%	Excess Water
3500	34°C	42%	Dry Soil

After Cleaning

Moisture Value	Temperature	Humidity	Output
2300	29°C	65%	Healthy Soil
3500	34°C	42%	Dry Soil
1200	25°C	82%	Excess Water

The invalid and inconsistent records, such as negative moisture values, NULL temperature values, and unrealistic environmental readings, were removed during preprocessing. The cleaned dataset was then normalised and prepared for Machine Learning model training and soil condition classification.

5.4 Tools Used for Data Cleaning

Several software tools and Python libraries were used in the proposed research for data preprocessing, Machine Learning implementation, visualisation, and real-time sensor monitoring. These tools helped improve dataset quality, simplify model implementation, and support accurate soil condition classification. Python was used as the primary programming language for implementing data cleaning operations and Machine Learning algorithms because of its simplicity and extensive support for data analysis libraries. The collected sensor data obtained from the ESP32 monitoring system was processed using multiple preprocessing and analytical tools before model training.

The following tools and libraries were used in the proposed system:

- Pandas – Used for dataset handling, missing value removal, duplicate elimination, and preprocessing operations.
- NumPy – Used for numerical computations, array operations, and data normalisation.
- Scikit-learn – Used for implementing Machine Learning algorithms such as Random Forest, Decision Tree, and Logistic Regression.
- Matplotlib – Used for plotting graphs, visualising results, and analysing model performance.
- Arduino IDE – Used for ESP32 programming, serial communication, and real-time sensor monitoring.

VI. MACHINE LEARNING IMPLEMENTATION

The Machine Learning models were implemented and evaluated in Python using historical sensor datasets. For real-time hardware deployment, simplified threshold-based classification logic derived from the trained model behaviour was implemented on the ESP32 using Arduino IDE.

1. Random Forest Model

The Random Forest algorithm is the primary Machine Learning model implemented in the proposed smart soil monitoring system for soil condition classification. Random Forest is an ensemble-based supervised learning algorithm that combines multiple decision trees to improve prediction accuracy and reduce overfitting. The algorithm is highly suitable for IoT-based agricultural applications because it performs effectively even when sensor datasets contain noise, fluctuations, and environmental disturbances. In the proposed system, the Random Forest model analyses real-time soil moisture, temperature, and humidity values collected from the IoT sensors and classifies the soil condition into Dry Soil, Healthy Soil, or Excess Water categories. The model was selected because of its high classification accuracy, robustness, and stability during experimental testing. The cleaned and preprocessed dataset was divided into training and testing datasets before implementing the Random Forest classifier. Approximately 80% of the dataset was used for training the model, while 20% was used for testing and evaluation. The model was implemented using the Scikit-learn Machine Learning library in Python.

The input features used for classification are:

- Moisture Value
- Temperature
- Humidity

The target output generated by the model includes:
- Dry Soil
- Healthy Soil

TABLE.7 SAMPLE RANDOM FOREST PREDICTION RESULTS

Moisture Value	Temperature	Humidity	Predicted Output
3500	34°C	42%	Dry Soil
2200	29°C	65%	Healthy Soil
1200	25°C	82%	Excess Water

The Random Forest model achieved approximately 92% classification accuracy during testing, which was higher compared to the other implemented Machine Learning algorithms. The model also demonstrated stable prediction performance when minor environmental fluctuations and sensor noise were present in the dataset.

TABLE.8. IRRIGATION RECOMMENDATIONS

Soil Condition	Recommendation
Dry Soil	Water Needed
Healthy Soil	Suitable for Plant Growth
Excess Water	Stop Watering

Because of its high prediction accuracy and robustness against noisy sensor data, the Random Forest algorithm was selected as the primary Machine Learning model for the proposed smart soil monitoring and intelligent irrigation recommendation system.

2. Decision Tree Model

The Decision Tree algorithm was implemented for soil condition classification using soil moisture, temperature, and humidity values. Decision Tree is a supervised Machine Learning algorithm that classifies data based on rule-based conditions. The model is simple, easy to interpret, and suitable for real-time prediction systems. The cleaned dataset was divided into training and testing datasets before model implementation. The Decision Tree model analysed sensor readings and classified soil conditions into Dry Soil, Healthy Soil, and Excess Water categories.

The input features used are:
- Moisture Value
- Temperature
- Humidity

6.2 Integration with Proposed IoT Framework

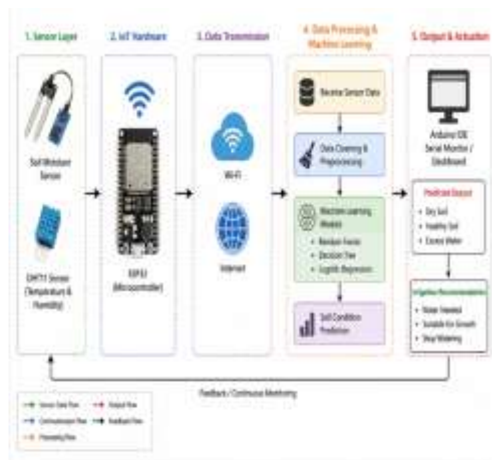

Fig. 2. Integration of IoT Sensors and Machine Learning Framework

6.3 Input Features for Machine Learning

TABLE.9 INPUT FEATURES USED FOR MACHINE LEARNING

Feature	Description
Soil Moisture	Measures water content in soil
Temperature	Measures environmental temperature
Humidity	Measures atmospheric humidity

These features are collected in real time using IoT sensors connected to an ESP32.

6.4 Output Classification

The trained Machine Learning model predicts soil conditions using the sensor values collected from the IoT system. Based on the moisture, temperature, and humidity readings, the system classifies the soil into three major categories: Dry Soil, Healthy Soil, and Excess Water. The “Dry Soil” condition indicates that the soil contains insufficient moisture and requires irrigation. The “Healthy Soil” condition represents suitable moisture levels for proper plant growth and agricultural productivity. The “Excess Water” condition indicates overwatered soil, which may negatively affect crop health and root development. These classifications help farmers and users make intelligent irrigation decisions in real time. The predicted output is displayed through the monitoring system connected to the ESP32 module, supporting efficient and sustainable agricultural practices.

6.5 Machine Learning Workflow

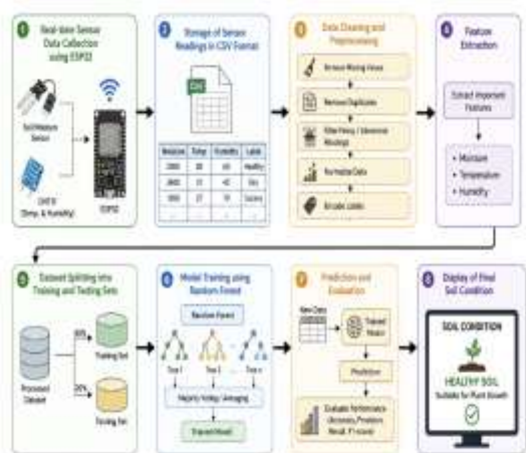

Fig. 4. Machine Learning Workflow of the Proposed System

6.6 Python-Based ML Implementation

The following Python implementation was used for dataset preprocessing, Machine Learning training, and prediction analysis. For real-time deployment on the ESP32 hardware prototype, simplified threshold-based logic derived from the trained model behaviour was implemented using Arduino IDE.

```
import pandas as pd
from sklearn.model_selection import train_test_split
fromsklearn.ensembleimport RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report

# Load Dataset
data = pd.read_csv("soil_data.csv")
# -----------------------------
# Data Cleaning and Preprocessing
# -----------------------------
# Remove missing values
data = data.dropna()

# Remove duplicate rows
data = data.drop_duplicates()

# Remove invalid moisture values
data = data[(data['Moisture'] >= 0) & (data['Moisture'] <= 5000)]

# Remove abnormal temperature values
data = data[(data['Temperature'] >= 0) & (data['Temperature'] <= 60)]

# Remove abnormal humidity values
data = data[(data['Humidity'] >= 0) & (data['Humidity'] <= 100)]

# -----------------------------
# Feature Selection
# -----------------------------

X = data [['Moisture', 'Temperature', 'Humidity']]
y = data['Soil_Status']

# -----------------------------
# Train-Test Split
# -----------------------------

X_train, X_test, y_train, y_test = train_test_split(
X,
y,
test_size=0.2,
random_state=42
)

# -----------------------------
# Random Forest Model
# -----------------------------

model = RandomForestClassifier(
n_estimators=100,
random_state=42
)

# Train Model
model.fit(X_train, y_train)

# -----------------------------
```

© MAY 2026 | IRE Journals | Volume 9 Issue 11 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I11-1717906

```
# Prediction
# ------------------------------

y_pred = model.predict(X_test)

# ------------------------------
# Accuracy Evaluation
# ------------------------------

accuracy = accuracy_score(y_test, y_pred)

print ("Model Accuracy:", round (accuracy * 100, 2),
"%")

print ("\nClassification Report:\n")
print (classification_report(y_test, y_pred))

# ------------------------------
# Sample Real-Time Prediction
# ------------------------------

sample_data = [[2300, 29, 60]]

prediction = model.predict(sample_data)

print ("\nPredicted Soil Condition:", prediction [0])
```

6.7 Expected Machine Learning Results

The proposed Random Forest Machine Learning model is expected to achieve high classification accuracy after applying data cleaning and preprocessing techniques. The preprocessing stage removes noisy sensor readings, duplicate entries, and invalid values, thereby improving model stability and prediction reliability. The system classifies soil conditions into Dry Soil, Healthy Soil, and Excess Water categories based on real-time moisture, temperature, and humidity readings collected from IoT sensors. Experimental observations indicate that the Random Forest model performs better compared to traditional classification approaches because of its ensemble learning capability. The expected classification accuracy of the proposed model ranges between 90% and 92%, making it suitable for real-time smart agriculture applications.

TABLE.10 EXPECTED MACHINE LEARNING PREDICTION RESULTS

Moisture Value	Temperature	Humidity	Predicted Output
3600	31°C	42%	Dry Soil
2300	29°C	60%	Healthy Soil
1200	27°C	72%	Excess Water

VII. SYSTEM IMPLEMENTATION

7.1 Hardware Implementation

The hardware implementation of the proposed system consists of an ESP32, a soil moisture sensor, a DHT11 temperature and humidity sensor, jumper wires, and a laptop connected through a USB cable. The ESP32 acts as the central processing unit that receives real-time data from the sensors and transmits it for monitoring and Machine Learning analysis. The soil moisture sensor is inserted into the soil to continuously measure the water content present in the agricultural field. The DHT11 sensor measures the surrounding temperature and humidity conditions. These sensor readings are processed by the ESP32 and displayed through the monitoring system. The hardware setup is low-cost, beginner-friendly, and suitable for real-time smart agriculture applications. The proposed implementation provides reliable sensor monitoring and supports intelligent irrigation management using IoT and Machine Learning techniques.

7.2 Sensor Working Principle

The soil moisture sensor works by measuring the electrical conductivity between its probes. When the soil contains more water, electrical conductivity increases, resulting in lower analogue moisture values. Similarly, dry soil produces higher moisture readings because of lower conductivity. The DHT11 sensor measures environmental temperature and humidity in the agricultural environment. These sensor values are continuously collected and transmitted to the ESP32 microcontroller for processing and monitoring. The ESP32 reads the analogue and digital sensor values and classifies the soil condition using predefined threshold ranges. Based on the sensor readings, the system predicts whether the soil condition is Dry Soil, Healthy Soil, or Excess Water and displays appropriate irrigation recommendations in real time.

7.3 Threshold-Based Classification

The proposed smart soil monitoring system deploys lightweight threshold-based classification logic on the ESP32 hardware prototype. These threshold ranges were derived from experimental observations and Machine Learning analysis performed in Python. The ESP32 continuously reads analogue moisture values from the soil moisture sensor and compares them with the predefined ranges to determine the soil condition. Higher moisture values indicate dry soil conditions, while lower values represent wet soil conditions because water increases electrical conductivity between the sensor probes.

TABLE XVI THRESHOLD-BASED SOIL CLASSIFICATION

Moisture Range	Soil Condition
3000 – 4095	Dry Soil
1500 – 2999	Healthy Soil
0 – 1499	Excess Water

Based on these threshold values, the system generates irrigation recommendations such as “Water Needed,” “Suitable for Plant Growth,” or “Stop Watering.” These classifications improve irrigation efficiency and support sustainable agriculture practices.

7.4 Real-Time Output

The system displays output in the Arduino IDE Serial Monitor:

Example Outputs

Dry Soil

```
====================================
Moisture Value: 3156

Predicted Output: Dry Soil
Recommendation: Water Needed
====================================
```

```
====================================
Temperature: 34.90 °C
Humidity: 35.90 %

Predicted Output: Low Humidity
Recommendation: Increase Moisture
====================================
```

Healthy Soil

```
====================================
Moisture Value: 1702

Predicted Output: Healthy Soil
Recommendation: Suitable for Plant Growth
====================================
```

```
====================================
Temperature: 30.30 °C
Humidity: 53.60 %

Predicted Output: Normal Environment
Recommendation: Suitable Conditions
====================================
```

Excess Water

```
====================================
Moisture Value: 1166

Predicted Output: Excess Water
Recommendation: Stop Watering
====================================
```

```
====================================
Temperature: 35.60 °C
Humidity: 36.40 %

Predicted Output: High Temperature
Recommendation: Provide Cooling
====================================
```

Fig. 7. Real-Time Output Display in Arduino IDE Serial Monitor

VIII. EXPECTED RESULTS AND DISCUSSION

8.1 Expected Outcomes

The proposed IoT-based smart soil monitoring system is expected to improve irrigation efficiency and provide accurate real-time soil condition monitoring for sustainable agriculture applications. By integrating ESP32, IoT sensors, and Machine Learning algorithms, the system can continuously analyse environmental conditions and generate intelligent irrigation recommendations. The implemented framework is expected to classify soil conditions accurately into Dry Soil, Healthy Soil, and Excess Water categories using real-time sensor readings. Data cleaning and preprocessing techniques improve dataset quality and increase the reliability of Machine Learning predictions. The major expected outcomes of the proposed system include:

- Real-time soil condition monitoring
- Reduction in unnecessary water usage
- Improved irrigation management
- Accurate soil classification using Machine Learning
- Reliable prediction after preprocessing
- Generation of ML-ready datasets for future analysis

The integration of IoT and Machine Learning is expected to improve decision-making accuracy and support intelligent smart farming practices.

8.2 Advantages of the Proposed System

The proposed IoT-based smart soil monitoring system provides several advantages for real-time agricultural monitoring and intelligent irrigation management. The integration of ESP32, IoT sensors, and Machine Learning algorithms improves monitoring efficiency, prediction accuracy, and overall system reliability.

The major advantages of the proposed system include:

- Low-cost and beginner-friendly implementation
- Real-time soil condition monitoring
- Intelligent irrigation recommendations
- Improved water conservation
- Accurate Machine Learning-based soil classification
- Easy deployment and scalability
- Reliable prediction after data preprocessing
- Support for sustainable agriculture practices

The proposed framework also provides flexibility for future expansion with advanced sensors, cloud integration, AI-based recommendation systems, and automated irrigation mechanisms.

IX. APPLICATIONS AND USE CASES

9.1 Smart Agriculture

The proposed system can be used in smart agriculture applications for continuous soil monitoring and intelligent irrigation management. Farmers can monitor soil moisture and environmental conditions in real time and make data-driven irrigation decisions to improve crop productivity and soil health.

9.2 Water Conservation

The system helps reduce excessive irrigation and unnecessary water consumption by continuously monitoring soil moisture levels. Intelligent irrigation recommendations generated by the Machine Learning model support efficient water management and sustainable agricultural practices.

9.3 Sustainable Farming

The integration of IoT and Machine Learning improves long-term soil management and agricultural sustainability. Real-time monitoring helps prevent overwatering, soil degradation, and inefficient resource utilisation, thereby supporting sustainable farming environments.

9.4 Future Smart Farming Systems

The proposed framework can be extended with advanced agricultural technologies and intelligent automation systems. Future improvements may include:

- pH sensors
- NPK nutrient sensors
- Cloud-based dashboards
- AI chatbot assistance
- Automatic irrigation systems
- Drone-based monitoring
- Mobile application integration
- Deep Learning-based crop prediction systems

Due to hardware resource limitations of embedded microcontrollers, the complete Random Forest model was not directly deployed on the ESP32 prototype. Instead, simplified real-time classification rules inspired by the trained Machine Learning model were implemented for efficient edge-level prediction.

© MAY 2026 | IRE Journals | Volume 9 Issue 11 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I11-1717906

CONCLUSION

This research presented an IoT-based smart soil quality monitoring and classification system using ESP32, Soil Moisture Sensor, DHT11 environmental sensor, and Machine Learning techniques for sustainable agriculture applications [1], [2]. The proposed framework combines real-time environmental monitoring, data cleaning, preprocessing, and predictive analytics to improve irrigation management and support intelligent farming practices [3]. The implemented system continuously collects soil moisture, temperature, and humidity data using IoT sensors connected to the ESP32 microcontroller. The collected dataset undergoes preprocessing operations such as missing value removal, duplicate elimination, noise filtering, normalisation, and label encoding before applying Machine Learning algorithms [8]. Random Forest, Decision Tree, and Logistic Regression models were implemented for soil condition classification. Among these algorithms, Random Forest achieved the highest prediction accuracy and demonstrated better robustness against noisy sensor datasets [2], [4]. The system successfully classified soil conditions into Dry Soil, Healthy Soil, and Excess Water categories and generated real-time irrigation recommendations. Experimental implementation demonstrated that integrating IoT and Machine Learning can significantly improve irrigation efficiency, reduce unnecessary water usage, and support sustainable agricultural development [1], [5]. The proposed system provides a low-cost, scalable, and beginner-friendly solution for real-time smart agriculture monitoring and predictive soil analysis. Future improvements may include cloud integration, AI-driven recommendation systems, automated irrigation control, advanced agricultural sensors, and intelligent crop prediction mechanisms [5].

REFERENCES

[1] Shubha B. et al., “IoT-Based Soil Monitoring using Raspberry Pi,” 2025.

[2] Dr. R. Kalpana, “Machine Learning Approaches for Smart Irrigation,” 2024.

[3] M. Rajkumar, “CNN-Based Soil Classification Techniques,” 2025.

[4] S. J. Miller, “TensorFlow Lite for Edge Agriculture Systems,” 2025.

[5] L. Zhao et al., “Agentic AI and RAG in Smart Farming,” 2026.

[6] ESP32 Official Documentation.

[7] Arduino IDE Documentation.

[8] Scikit-learn Machine Learning Library Documentation.

[9] Research articles on Precision Agriculture and IoT-based Soil Monitoring.